# Creative Coding for the Web

## Session 02
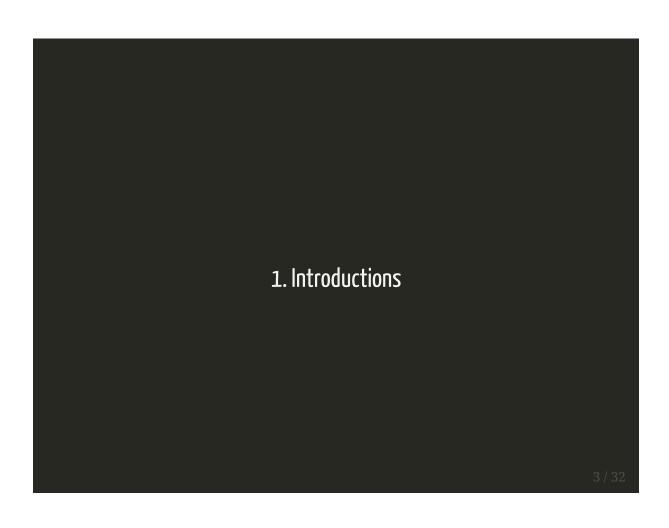
09.04.15

# Agenda

1. Introductions

2. Syllabus Review

3. Overview:

   - The web
   - This course
   - Expectations (yours and mine)

4. Technical Details

5. Workshop

6. Research Methods

# 1. Introductions

# Hello

## Bio

Call me ... Rhazes

I am a ... *old* studying ... *technology, science, math, art, gaming*

# Hello

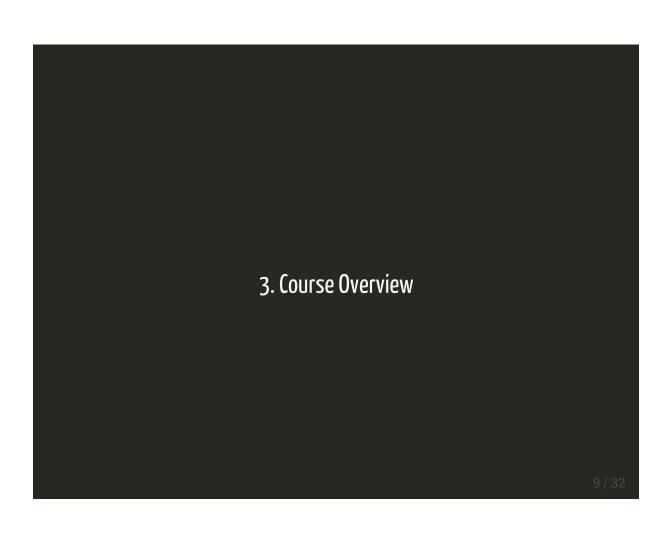## Bio

Call me ...

I am a ... studying ...

# Hello

## Skillz

- I am THE [wo]man in:
  (Design, HTML, CSS, JS, the command line)

- I've got some chops in:
  (Design, HTML, CSS, JS, the command line)

- I'm decent in:
  (Design, HTML, CSS, JS, the command line)

- I've done a little:
  (Design, HTML, CSS, JS, the command line)

- I've never heard of:
  (Design, HTML, CSS, JS, the command line)

- I am [not] familiar with these technologies
  (templating enginges, sass/less, markdown, APIs)

# Hello

## Goals

- When I grow up I want to …
- My goals for this class are …
- I have always wondered about …

# 2. Syllabus Review

# 3. Course Overview

# What is this class about?

The web is an integral part of contemporary culture. It is a foundation for entertainment, connection, communication, social action, commerce, and descension. As members of society we are automatically participants in the web. As media designers and artists we can be more than consumers of digital culture we can be be producers, innovators and critics.

The good news is that the path to acquire these skills is accessible to anyone with an interest and the desire to learn. In addition to making you more marketable for jobs these skills and this knowledge can you help you **create** your own jobs.

# Course objectives

- Become familiar with contemporary web technologies
  Contemporary Web

- Think critically about the internet as a tool for
  intervention and change

- Learn and enhance your skills as front-end developers
  and designers. (i.e. creative technologists)

- Have a play with your ideas and concepts

# Expectations

Technology is a contact sport. You can only learn by doing so I expect participation and effort. The faster you can run into problems the faster your skills will grow.

- Don't get frustrated

- You are not not-smart, untechnical, non-mathy, non-coder ....

- Anyone can learn this stuff and plenty have

4. Let's Get Started ...

DubDubDub

**what is the web?**

## DubDubDub

## what is the web?

Imagine if we gave everyone in the world:

a monitor ...

a keyboard ...

3 USB thumb drives ...

DubDubDub

what is the
web?

... and all over the world we installed adaptors for our USB sticks



Dead Drops

# The Web ...

... is a collection of connected file directories that remote user's can access (create, read, update and delete resources)

# Web Architecture ...

.. the design of the structure of a set of networks, a particular network or a single website.

Web architecture can loosely be divided into 2 main classes:

1. Front End
2. Back End

# Web Architecture ...

1. Client request resources **(Front End)**

2. "Internet plumbing" take those request and find out which computer has the requested files

3. The computer that has the file "serves" the client's "request" **(Back End)**

4. The client computer has a program that can interpret what to do with the file that was requested (display an image, play an audio file, etc. ) **(Front End)**

# Web Architecture ...

1. Client request resources **(Front End)**

2. "Internet plumbing" take those request and find out which computer has the requested files

3. The computer that has the file "serves" the client's "request" **(Back End)**

4. The client computer has a program that can interpret what to do with the file that was requested (display an image, play an audio file, etc. ) **(Front End)**

*we will have to do a little back end work (step 3) to get our resources onto the server*

## In this class our concerns are...

## 1. Client requests resources

We will investigate:

- Standard resources requests
- In page requests
- Accessing APIs

In this class
our
concerns
are…

## 4. The client computer interprets the resource returned from the server

- Effective design strategies for presentation
- Static presentation of content
- Dynamic presentation of content
- Design strategies for dynamic presentation and interaction

# Web Architecture ...

Another characteristic of web architecture is "Separation of Concerns"

## Separation of concerns

is the idea that each component in your system is responsible for performing one task. This allows for a "kit of parts".

## A kit of parts

supports the concept of reuse: resusing components in different combimations to create custom solutions to problems.

# Separation of Concerns

## Front End Concerns

- **Content**
  what is the information that you want to display

- **Presentation**
  how should the content be styled

- **Interaction**
  how should the content and presentation change
  when the user does things

# Separation of Concerns

## Front End Concerns

- **Content**
  what is the information that you want to display
  *Hyper Text Markup Language (HTML)*

- **Presentation**
  how should the content be styled
  *Cascading Style Sheets (CSS)*

- **Interaction**
  how should the content and presentation change
  when the user does things
  *Javascript (JS)*

# Separation of Concerns

Example,

## Remark JS

Remark is a web based slide presentation framework. Write your presentation in Markdown and serve it on the web in a custom application.

## Development Tools

1. A server to serve files; tools for managing the file system
   *storm*; *unix*

2. Tools to create the files
   any text editor, *sublime*, *light table*

3. Tools for troubleshooting
   *chrome developer tools*

4. Tools for managing files and collaboration
   *github*
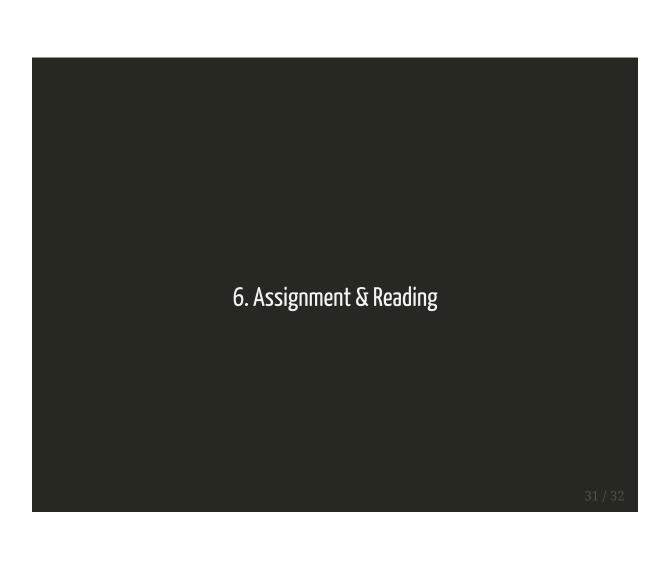
# 5. Workshop

# Build a simple webpage about your favorite animal

Can't think of an animal??

Your spirit animal

## Development Tools

1. A server to serve files

2. Tools to create the files

3. Tools for troubleshooting

4. Tools for managing files and collaboration

# 6. Assignment & Reading

# For next week

## Reading

"As We May Think" by Vannevar Bush

## Work through

1. Code Academy HTML & CSS
2. Markdown Tutorial

### extra

- Code Academy Javascript
  stop after Arrays section.

- Also take a look at Stack Edit. They have a good
  example of Markdown as the default page on their
  editor.